

FIȘA DISCIPLINEI¹⁾

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea Petrol-Gaze din Ploiești
1.2. Facultatea	Litere și Științe
1.3. Departamentul	Informatică, Tehnologia Informației, Matematică și Fizică
1.4. Domeniul de studii universitare	Informatică
1.5. Ciclul de studii universitare	Licență
1.6. Programul de studii universitare	Informatică

2. Date despre disciplină

2.1. Denumirea disciplinei	Metode avansate de programare
2.2. Titularul activităților de curs	Lector dr. Tudorică Daniela
2.3. Titularul activităților aplicative	Lector dr. Tudorică Daniela
2.4. Anul de studiu	I
2.5. Semestrul *	2
2.6. Tipul de evaluare	Examen
2.7. Categoria formativă** / regimul*** disciplinei	S2/ O

* numărul semestrului este conform planului de învățământ;

** fundamentală = F0; de domeniu = D1; de specialitate = S2; complementară = C3

*** obligatorie = O; opțională = A; facultativă = L

3. Timpul total estimat (ore pe semestru al activităților didactice)

3.1. Număr de ore pe săptămână	5	din care:	2	3.3. Seminar/laborator	3
		3.2. curs			
3.4. Total ore din planul de învățământ	70	din care:	28	3.6. Seminar/laborator	42
		3.5. curs			
3.7. Distribuția fondului de timp					ore
Studiu după manual, suport de curs, bibliografie și notițe					30
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					30
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					10
Tutoriat					5
Examinări					5
Alte activități					0
3.7. Total ore studiu individual	80				
3.8. Total ore pe semestru	150				
3.9. Numărul de credite	6				

4. Precondiții (acolo unde este cazul)

4.1. de curriculum	➤ Fundamentele programării, Algoritmi fundamentali
4.2. de competențe	➤ Proiectarea algoritmilor, cunoștințe de programare în limbajul C

¹⁾ Adaptare după Ordinul Ministrului educației, cercetării, tineretului și sportului nr. 5 703/2011 privind implementarea Codului național al calificărilor din învățământul superior, publicat în Monitorul Oficial al României, partea I, nr.880 bis / 13.XII.2011

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	<ul style="list-style-type: none"> sală de curs multimedia pentru realizarea de prelegeri, conversații, dezbateri
5.2. de desfășurare a seminarului/laboratorului	<ul style="list-style-type: none"> laborator dotat cu sisteme de calcul cu mediu integrat de dezvoltare instalat (Code Blocks) pentru scrierea de programe în limbajul C++

6. Competențe specifice acumulate

Competențe profesionale	<p>C1.2 Explicarea unor aplicații soft existente, pe niveluri de abstracție (arhitectură, pachete, clase, metode) utilizând în mod adecvat cunoștințele de bază.</p> <p>C2.1 Identificarea de metodologii adecvate de dezvoltare a sistemelor software.</p> <p>C2.2 Identificarea și explicarea mecanismelor adecvate de specificare a sistemelor software.</p> <p>C3.2 Identificarea și explicarea modelelor informatice de bază adecvate domeniului de aplicare.</p> <p>C2.3 Utilizarea metodologiilor, mecanismelor de specificare și a mediilor de dezvoltare pentru realizarea aplicațiilor informatice.</p> <p>C3.3 Utilizarea modelelor și instrumentelor informatice și matematice pentru rezolvarea problemelor specifice domeniului de aplicare.</p>
Competențe transversale	<p>CT1. Aplicarea regulilor de muncă organizată și eficientă, a unor atitudini responsabile față de domeniul didactic-științific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor și a normelor de etică profesională</p> <p>CT2. Desfășurarea eficientă a activităților organizate într-un grup inter-disciplinar și dezvoltarea capacităților empatice de comunicare inter-personală, de relaționare și colaborare cu grupuri diverse</p>

7. Obiectivele disciplinei (reieșind din grila competențelor specifice acumulate)

7.1. Obiectivul general al disciplinei	Formarea de competențe profesionale și transversale necesare obținerii calificării. Obiectivul principal al disciplinei constă în familiarizarea studenților cu fundamentele teoretice și aplicative ale metodelor avansate de programare în limbajul C++.
7.2. Obiectivele specifice	<p>Formarea competențelor profesionale (C1.2, C2.1, C2.2, C3.2, C2.3, C3.3) și transversale (CT1, CT2).</p> <p>La finalul activităților, studentul va fi capabil:</p> <ul style="list-style-type: none"> ➤ să elaboreze algoritmi de rezolvare a unor probleme practice; ➤ să codifice algoritmi în limbajul C++, utilizând facilitățile noilor standarde (C++14, C++17, C++20); ➤ să utilizeze un mediu integrat de dezvoltare (Code Blocks) pentru scrierea de programe în limbajul C++; ➤ să aplice reguli de muncă organizată și eficientă, să manifeste atitudini responsabile față de domeniul didactic-științific, să respecte principiile și normele de etică profesională, să lucreze în echipă.

8. Conținuturi

8.1. Curs	Nr.ore	Metode de predare	Observații
1. Prezentarea obiectivelor disciplinei, a modului de desfășurare a orelor de curs și de laborator, prezentarea modului de evaluare. Recapitulare fundamentele programării: etapele rezolvării unei probleme cu ajutorul calculatorului, algoritmi. 2. Limbajul C++. Generalități și scurt istoric, structura unui program C++, biblioteci C++, spații de nume (namespace std), operații de citire/scriere, tipuri de date. 3. Operatori, tablouri, structuri, pointeri și referințe în C++. 4. Lucrul cu string-uri în C++. 5. Lucrul cu fișiere în C++. 6. Funcții în C++. 7. Programare avansată în C++. Directive preprocesor. Tratarea excepțiilor 8. Funcții template. Clase template. Iteratori. Containere 9. Grafică în C++. 10. Realizarea de interfețe grafice (GUI) pentru aplicațiile C++ 11. Aplicații C++ cu mai multe fișiere 12. Biblioteci C++: <i>algorithm</i> , <i>random</i> , <i>sstream</i> etc. 13. Recapitulare.	1. 2 ore 2. 2 ore 3. 4 ore 4. 2 ore 5. 2 ore 6. 2 ore 7. 2 ore 8. 2 ore 9. 2 ore 10. 2 ore 11. 2 ore 12. 2 ore 13. 2 ore	Expunere, studii de caz, conversație, dezbateri, utilizare tehnologii multimedia	
Bibliografie <ol style="list-style-type: none"> Alexandrescu A., <i>Modern C++ Design</i>, Addison-Wesley, 2001 Andonie R, Gârbacea I., <i>Algoritmi fundamentali. O perspectivă C++</i>, Ed. Libris, Cluj-Napoca, 1995 Coșulski M., <i>Algoritmi fundamentali. Proiectare și implementare</i>, Ed. Universitaria, Craiova, 2015 Levitin A., <i>Introduction to the design and analysis of algorithms</i>, Pearson Education, third edition, 2012 Marinoiu Cr, <i>Programarea în limbajul C</i>, Editura Universității din Ploiești, 2000 Stroustrup B., <i>The C++ Programming Language</i>, Addison-Wesley, 1985 Stroustrup B., <i>Programming: Principles and Practice Using C++</i>, Second Edition, Addison-Wesley, 2014 Manual online de C++ (CPlusPlus): http://www.cplusplus.com/doc/tutorial/ Curs online de C++ (LearnCpp): http://www.learncpp.com/ Curs online de C++ (TutorialsPoint): http://www.tutorialspoint.com/cplusplus/index.htm 			
8.2. Seminar / laborator/proiect	Nr. ore	Metode de predare	Observații
1. Recapitulare – Fundamentele programării: declararea și inițializarea variabilelor, instrucțiuni pentru codificarea operațiilor de bază și a structurilor de control dintr-un algoritm 2. Aplicații cu tablouri (vectori și matrice) și pointeri 3. Aplicații cu structuri 4. Aplicații cu stringuri (clasa <i>string</i>) 5. Aplicații cu fișiere (<i>fstream</i>) 6. Funcții definite de utilizator 7. Funcții recursive 8. Directive preprocesor. Tratarea excepțiilor 9. Definierea și utilizarea funcțiilor template. Lucrul cu containere: <i>array</i> , <i>vector</i>	1. 2 ore 2. 4 ore 3. 4 ore 4. 4 ore 5. 4 ore 6. 4 ore 7. 2 ore 8. 2 ore 9. 2 ore 10. 4 ore 11. 4 ore 12. 4 ore 13. 2 ore	Dezbateri, studii de caz, exemple, angajarea fiecărui student în descoperirea cunoștințelor.	

10. Grafică în C++ 14. Interfețe grafice (GUI) în C++ 15. Utilizarea bibliotecilor <i>algorithm</i> , <i>random</i> , <i>sstream</i> etc. 16. Evaluare laborator			
Bibliografie <ol style="list-style-type: none"> Alexandrescu A., <i>Modern C++ Design</i>, Addison-Wesley, 2001 Andonie R, Gârbacea I., <i>Algoritmi fundamentali. O perspectivă C++</i>, Ed. Libris, Cluj-Napoca, 1995 Stroustrup B., <i>The C++ Programming Language</i>, Addison-Wesley, 1985 Stroustrup B., <i>Programming: Principles and Practice Using C++</i>, Second Edition, Addison-Wesley, 2014 Manual online de C++ (CPlusPlus): http://www.cplusplus.com/doc/tutorial/ Curs online de C++ (LearnCpp): http://www.learncpp.com/ Curs online de C++ (TutorialsPoint): http://www.tutorialspoint.com/cplusplus/index.htm 			

9. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatori reprezentativi din domeniul aferent programului

- Cursul și laboratorul sunt astfel concepute încât, prin competențele formate, să răspundă cerințelor pieței muncii.
- Conținutul acestei discipline oferă cursanților cunoștințe de proiectare și implementare a aplicațiilor C++, cunoștințe indispensabile în cazul în care absolventul decide să urmeze o carieră în domeniul software.

10. Evaluare

Tip activitate	10.1. Criterii de evaluare	10.2. Metode de evaluare	10.3. Pondere din nota finală
10.4. Curs	Completitudinea și corectitudinea cunoștințelor, capacitatea de a opera cu cunoștințele asimilate, capacitatea de analiză, de interpretare personală, gradul de asimilare a limbajului de specialitate și capacitatea de comunicare.	Examen scris	50% Pentru promovarea examenului minim nota 5 la proba scrisă.
10.5. Seminar/laborator/proiect	Aplicarea practică a elementelor prezentate la orele de curs și de laborator, prin realizarea de aplicații în limbajul C++.	Evaluare orală	50% Pentru promovarea examenului minim nota 5 la laborator.
10.6. Standard minim de performanță			
Asimilarea limbajului de specialitate privind programarea avansată în C++. Realizarea unor programe C++ pentru rezolvarea unor probleme de dificultate medie.			

Data completării
20.09.2020

Semnătura titularului de curs
Lector dr. Daniela Tudorică

Semnătura titularului de seminar/laborator
Lector dr. Daniela Tudorică

Data avizării în departament

Semnătura directorului de departament
Conf. dr. Gabriela Moise